UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/717,822	11/20/2003	James Chien-Chiung Chen	TUC920030129US1	8660

36491 7590 02/22/2007
KUNZLER & ASSOCIATES
8 EAST BROADWAY
SUITE 600
SALT LAKE CITY, UT 84111

EXAMINER	
BROWN, MICHAEL J	
ART UNIT	**PAPER NUMBER**
2116	

SHORTENED STATUTORY PERIOD OF RESPONSE	MAIL DATE	DELIVERY MODE
3 MONTHS	02/22/2007	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

If NO period for reply is specified above, the maximum statutory period will apply and will expire 6 MONTHS from the mailing date of this communication.

Office Action Summary	Application No.	Applicant(s)	
	10/717,822	CHEN ET AL.	
	Examiner	Art Unit	
	Michael J. Brown	2116	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *07 February 2007*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1,3-16,19-24 and 27-37* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☐ Claim(s) *1,3-16,19-24 and 27-37* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *20 November 2003* is/are: a)☒ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

DETAILED ACTION

Claim Objections

1. Claims 13 and 14 are objected to because of the following informalities: Claims 13 and 14 appear to be dependant upon claim 12 as opposed to claim 1. The "bootstrap module" is first introduced in claim 12 and not in claim 1. Both claim 13 and 14 call on the bootstrap module of claim 12. Appropriate correction is required.

Claim Rejections - 35 USC § 103

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

The factual inquiries set forth in *Graham* **v.** *John Deere Co.*, 383 U.S. 1, 148 USPQ 459 (1966), that are applied for establishing a background for determining obviousness under 35 U.S.C. 103(a) are summarized as follows:

1. Determining the scope and contents of the prior art.
2. Ascertaining the differences between the prior art and the claims at issue.
3. Resolving the level of ordinary skill in the pertinent art.
4. Considering objective evidence present in the application indicating obviousness or nonobviousness.

2. Claims 1, 4-16, 19-24, and 27-37 are rejected under 35 U.S.C. 103(a) as being unpatentable over Wu et al.(US Patent 6,732,267) and further in view of Hiller et al.(US Patent 6,658,659) and Talati et al.(US PGPub 2004/0044997).

As to claim 1, Wu discloses a concurrent code load apparatus for fastload code image update on a communications adapter(target system 102, see Fig. 1), the apparatus comprising an image load module(system administrator; see column 3, line 62) configured to load a copy of a new code image(updated BIOS image; see column 3, line 63) in a memory(storage device; see column 3, line 67) on the communications adapter, the memory concurrently storing a copy of an old code image(old system BIOS; see column 4, line 30) used by the communications adapter. Wu also discloses the apparatus comprising a memory initialization module(operating system module; see column 3, line 65) configured to invoke the new code image to perform a memory initialization operation(see column 3, lines 64-67), and an image overlay module configured to overlay the old code image with the new code image(see Fig 2, Item 216). However, Wu fails to disclose a query module configured to determine an incompatibility between the old code image and the new code image, and an image bridge module configured to reconcile the incompatibility between the old and new code image. Also Wu fails to specifically disclose these functions occurring while the old code image is still executing.

Hiller teaches a query module(compatibility vector 342, see Fig. 3C) configured to determine an incompatibility between the old code image and the new code image,

and an image bridge module(loader; see column 9, line 53) configured to reconcile the incompatibility between the old and new code image(see column 9, lines 50-54).

Talati teaches similar functions occurring while the old code image is still executing(see paragraph 0006). It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the teaches of Wu with the teaches of Hiller and Talati to include incompatibility detection and reconciliation, all while an old code image is still executing, to the Wu system. The motivation to do so would be to create a system that can check and ensure that loaded software modules are compatible with one another and will therefore execute properly(Hiller Abstract, lines 10-13) while achieving a non-disruptive code load(Talati Abstract, lines 1-2).

As to claim 4, Wu discloses the apparatus further comprising a fastload key module configured to create and store a fastload key to indicate a fastload code image update on the communications adapter(see column 4, lines 52-55).

As to claim 5, Wu discloses the apparatus further comprising a fastload adapter initialization module configured to initialize the communications adapter using a fastload initialization sequence in response to a fastload code image update(see column 4, lines 52-55).

As to claim 6, Wu discloses the apparatus wherein the fastload adapter initialization module is further configured to access a fastload key prior to using the fastload initialization sequence(see column 4, lines 52-55).

As to claim 7, Wu discloses the apparatus further comprising a standard adapter initialization module configured to initialize the communications adapter using a

standard initialization sequence in response to a failure to access a fastload key(see column 4, lines 52-55).

As to claim 8, Wu discloses the apparatus further comprising a fastload adapter initialization module configured to execute a fastload initialization operation during the standard initialization sequence(see column 4, lines 52-55).

As to claim 9, Wu discloses the apparatus wherein the old code image comprises a code image update module configured to control a code image update(see column 4, lines 30-32).

As to claim 10, Wu discloses the apparatus wherein the code image update module comprises a load module configured to load the new code image in the memory(see column 4, lines 30-32).

As to claim 11, Wu discloses the apparatus wherein the code image update module comprises a branch module configured to branch from the old code image to the new code image(see column 4, lines 30-32).

As to claim 12, Wu discloses the apparatus wherein the new code image comprises a bootstrap module configured to define a bootstrap operation, the bootstrap operation configured to facilitate a code image update(see column 4, lines 27-32).

As to claim 13, Hiller teaches the apparatus wherein the bootstrap module comprises the image bridge module(compatibility vector 342, see Fig. 3C).

As to claim 14, Hiller discloses the apparatus wherein the bootstrap module comprises the image overlay module(loader; see column 9, line 53). Wu discloses the

image overlay module configured to overlay the old code image with the new code image using the copy module(see column 4, lines 30-32).

As to claim 15, Wu discloses a storage system for facilitating fastload code image update on a source communications adapter(target system 102, see Fig. 1), the storage system comprising a source input device(system administrator; see column 3, line 62) configured to receive a source electronic storage media device(storage device; see column 3, line 67), the source electronic storage media device configured to store a new code image(updated BIOS image; see column 3, line 63). Wu also discloses the system comprising a storage system processor(operating system module; see column 3, line 65) configured to initiate the fastload code image update and notify the source communications adapter of the fastload code image update(see column 3, lines 64-67), and the source communications adapter configured to copy the new code image to a local memory device and to implement the fastload code image update(see Fig 2, Item 216). However, Wu fails to disclose the storage system processor further configured to determine and reconcile an incompatibility between the old code image and the new code image while the old code image is still executing.

Hiller teaches a system configured to determine and reconcile an incompatibility between an old code image and a new code image(see column 9, lines 50-54).

Talati teaches similar functions being conducted while the old code image is still executing(see paragraph 0006). It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the teaches of Wu with the teaches of Hiller and Talati to include incompatibility detection and reconciliation, all

while an old code image is still executing, to the Wu system. The motivation to do so would be to create a system that can check and ensure that loaded software modules are compatible with one another and will therefore execute properly(Hiller Abstract, lines 10-13) while achieving a non-disruptive code load(Talati Abstract, lines 1-2).

As to claim 16, Wu discloses a method for fastload code image update on a communications adapter(target system 102, see Fig. 1), the method comprising loading a copy of a new code image(updated BIOS image; see column 3, line 63) in a memory(storage device; see column 3, line 67) on the communications adapter, the memory concurrently storing a copy of an old code image(old system BIOS; see column 4, line 30) used by the communications adapter. Wu also discloses the method comprising invoking the new code image to perform a memory initialization operation(see column 3, lines 64-67), and overlaying the old code image with the new code image(see Fig 2, Item 216). However Wu fails to disclose the method determining an incompatibility between the old code image and the new code image, and reconciling the incompatibility between the old code image and the new code image while the old code image is still executing.

Hiller teaches a method determining an incompatibility between the old code image and the new code image, and reconciling the incompatibility between the old code image and the new code image(see column 9, lines 50-54).

Talati teaches similar function being conducted while the old code image is still executing(see paragraph 0006). It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the teaches of Wu with the

teaches of Hiller and Talati to include incompatibility detection and reconciliation, all while an old code image is still executing, to the Wu system. The motivation to do so would be to create a system that can check and ensure that loaded software modules are compatible with one another and will therefore execute properly(Hiller Abstract, lines 10-13) while achieving a non-disruptive code load(Talati Abstract, lines 1-2).

As to claim 19, Wu discloses the method further comprising creating and storing a fastload key to indicate a fastload code image update on the communications adapter(see column 4, lines 52-55).

As to claim 20 Wu discloses the method further comprising initializing the communications adapter using a fastload initialization sequence in response to a fastload code image update(see column 4, lines 52-55).

As to claim 21, Wu discloses the method further comprising determining if access a fastload key prior to using the fastload initialization sequence(see column 4, lines 52-55).

As to claim 22, Wu discloses the method further comprising initializing the communications adapter using a standard initialization sequence in response to a failure to access a fastload key(see column 4, lines 52-55).

As to claim 23, Wu discloses a method for fastload code image update on a communications adapter(target system 102, see Fig. 1), the method comprising loading a copy of a new code image(updated BIOS image; see column 3, line 63) in a memory(storage device; see column 3, line 67) on the communications adapter, the memory concurrently storing a copy of an old code image(old system BIOS; see column

4, line 30) used by the communications adapter. Wu also discloses the method comprising invoking the new code image to perform a memory initialization operation(see column 3, lines 64-67), and overlaying the old code image with the new code image(see Fig 2, Item 216), creating and storing a fastload key to indicate a fastload code image update on the communications adapter, and initializing the communications adapter using a fastload initialization sequence in response to a fastload code image update(see column 4, lines 52-55). However, Wu fails to disclose the method determining an incompatibility between the old code image and the new code image, and reconciling the incompatibility between the old code image and the new code image while the old code image is still executing.

Hiller teaches a method determining an incompatibility between the old code image and the new code image, and reconciling the incompatibility between the old code image and the new code image(see column 9, lines 50-54).

Talati teaches similar functions being conducted while the old code image is still executing(see paragraph 0006). It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the teaches of Wu with the teaches of Hiller and Talati to include incompatibility detection and reconciliation, all while an old code image is still executing, to the Wu system. The motivation to do so would be to create a system that can check and ensure that loaded software modules are compatible with one another and will therefore execute properly(Hiller Abstract, lines 10-13) while achieving a non-disruptive code load(Talati Abstract, lines 1-2).

As to claim 24, Wu discloses a computer readable storage medium comprising computer readable code configured to carry out a method for fastload code image update on a communications adapter(target system 102, see Fig. 1), the method comprising loading a copy of a new code image(updated BIOS image; see column 3, line 63) in a memory(storage device; see column 3, line 67) on the communications adapter, the memory concurrently storing a copy of an old code image(old system BIOS; see column 4, line 30) used by the communications adapter. Wu also discloses the computer readable storage medium comprising invoking the new code image to perform a memory initialization operation(see column 3, lines 64-67), and overlaying the old code image with the new code image(see Fig 2, Item 216). However Wu fails to disclose the method determining an incompatibility between the old code image and the new code image, and reconciling the incompatibility between the old code image and the new code image while the old code image is still executing.

Hiller teaches a method determining an incompatibility between the old code image and the new code image, and reconciling the incompatibility between the old code image and the new code image(see column 9, lines 50-54).

Talati teaches similar functions being conducted while the old code image is still executing(see paragraph 0006). It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the teaches of Wu with the teaches of Hiller and Talati to include incompatibility detection and reconciliation, all while an old code image is still executing, to the Wu system. The motivation to do so would be to create a system that can check and ensure that loaded software modules

are compatible with one another and will therefore execute properly(Hiller Abstract, lines 10-13) while achieving a non-disruptive code load(Talati Abstract, lines 1-2).

As to claim 27, Wu discloses the computer readable storage medium wherein the method further comprises creating and storing a fastload key to indicate a fastload code image update on the communications adapter(see column 4, lines 52-55).

As to claim 28, Wu discloses the computer readable storage medium wherein the method further comprises initializing the communications adapter using a fastload initialization sequence in response to a fastload code image update(see column 4, lines 52-55).

As to claim 29, Wu discloses the computer readable storage medium wherein the method further comprises determining if access a fastload key prior to using the fastload initialization sequence(see column 4, lines 52-55).

As to claim 30, Wu discloses the computer readable storage medium wherein the method further comprises initializing the communications adapter using a standard initialization sequence in response to a failure to access a fastload key(see column 4, lines 52-55).

As to claim 31, Wu discloses the computer readable storage medium wherein the old code image comprises a code image update module configured to control a code image update(see column 4, lines 30-32).

As to claim 32, Wu discloses the computer readable storage medium wherein the code image update module comprises a load module configured to load the new code image in the memory(see column 4, lines 30-32).

As to claim 33, Wu discloses the computer readable storage medium wherein the code image update module comprises a branch module configured to branch from the old code image to the new code image(see column 4, lines 30-32).

As to claim 34, Wu discloses the computer readable storage medium wherein the new code image comprises a bootstrap module configured to define a bootstrap operation, the bootstrap operation configured to facilitate a code image update(see column 4, lines 27-32).

As to claim 35, Wu discloses the computer readable storage medium wherein the bootstrap module comprises a conversion module, the image bridge module configured to reconcile an incompatibility between the old code image and the new code image using the conversion module(see column 4, lines 34-45).

As to claim 36, Wu discloses the computer readable storage medium wherein the bootstrap module comprises a copy module, the image overlay module configured to overlay the old code image with the new code image using the copy module(see column 4, lines 30-32).

As to claim 37, Wu discloses an apparatus for fastload code image update on a communications adapter(target system 102, see Fig. 1), the apparatus comprising means for loading(system administrator; see column 3, line 62) a copy of a new code image(updated BIOS image; see column 3, line 63) in a memory(storage device; see column 3, line 67) on the communications adapter, the memory concurrently storing a copy of an old code image(old system BIOS; see column 4, line 30) used by the communications adapter. Wu also discloses the apparatus comprising means for

invoking(operating system module; see column 3, line 65) the new code image to perform a memory initialization operation(see column 3, lines 64-67), and means for overlaying the old code image with the new code image(see Fig 2, Item 216). However, Wu fails to disclose the apparatus comprising means for determining an incompatibility between the old code image and the new code image, and means for reconciling the incompatibility between the old code image and the new code image while the old code image is still executing.

Hiller teaches an apparatus comprising means for determining an incompatibility between the old code image and the new code image(compatibility vector 342, see Fig. 3C), and means for reconciling the incompatibility between the old code image and the new code image(loader; see column 9, line 53)(see column 9, lines 50-54).

Talati teaches similar functions being conducted while the old code image is still executing(see paragraph 0006). It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the teaches of Wu with the teaches of Hiller and Talati to include incompatibility detection and reconciliation, all while an old code image is still executing, to the Wu system. The motivation to do so would be to create a system that can check and ensure that loaded software modules are compatible with one another and will therefore execute properly(Hiller Abstract, lines 10-13) while achieving a non-disruptive code load(Talati Abstract, lines 1-2).

Response to Arguments

3. Applicant's arguments, see Remarks, filed 2/7/2007, with respect to the rejection(s) of claim(s) 1-37 under Wu et al.(US Patent 6,732,267) have been fully considered and are persuasive. Therefore, the rejection has been withdrawn. However, upon further consideration, a new ground(s) of rejection is made in view of Wu et al.(US Patent 6,732,267) and further in view of Hiller et al.(US Patent 6,658,659) and Talati et al.(US PGPub 2004/0044997).

Conclusion

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Michael J. Brown whose telephone number is (571)272-5932. The examiner can normally be reached Monday-Thursday from 7:00am-5:30pm.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Rehana Perveen can be reached on (571)272-3676. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

Michael J. Brown
Art Unit 2116

REHANA PERVEEN
SUPERVISORY PATENT EXAMINER
2/20/07